AMENDMENTS TO THE SPECIFICATION

Please amend paragraph [0010] as follows:

There exists a need for effectively and holistically modeling and analyzing life cycle operations and support of missile and missile defense systems capturing the complexity, interdependencies, and random nature of the problem. Any approach should affordably provide accurate, reusable and portable models for analyzing the problem.

Please amend paragraph [0012] as follows:

This is accomplished by first creating a model of the O&S problem based on a service use profile (SUP) that describes a logical structure of delivery, maintenance, deployment and testing policy and infrastructure and logistics constraints. The model is translated into a discrete ~~even~~ event simulation in which dynamic objects (weapons) flow through a network of static objects that are organized in accordance with the model. The static objects are defined by data that is global with respect to the dynamic objects and functional operators at least some of which are probabilistic. Decisions are made based on the local state of a dynamic object or a global state of the network at least some of which are randomized in occurrence and/or time duration by the functional operators. Each dynamic object includes common attributes with unique deterministic or random values at least some of which are updated as the dynamic object flows through the network and decisions are made. The simulation outputs a time-based prediction of weapons availability, maintenance activities, and spare parts stock over a life cycle of the weapons system.

Please amend paragraph [0058] as follows:

The methodology for creating a comprehensive model that describes a particular missile or missile defense system and then translating that model into a valid and reliable discrete ~~even~~ event simulation is complicated. The sheer scale of weapons systems O&S analysis, considering the number and differing variants of weapons, the complexity of weapons tests, the number of different observations that must be made, the interdependency of those observations, the granularity as a function of time and environment, etc. requires a systematic approach for developing the model and DES. The definition of a SUP and the use of the common attributes and pre-validated common blocks and sub-models simplify the process and improves consistency. For each designer to use an ad hoc approach using only the primitives supported by a particular software package and redefining all static and dynamic objects on a case-by-case basis would be inefficient and induce inconsistency between modeling projects.

Please amend paragraph [0088] as follows:

Document and Implement Results. The objectives, ground rules and assumptions, experiments conducted, and results are recorded and presented to client(s) to ~~aide~~ aid their decisions.

Please amend paragraph [0104] as follows:

As shown in FIG. 9, the Stockpile Availability block 170 is implemented using a number of primitive blocks. The primitives used, the topology and the instruction set are determined by the simulation software and designer choice, other combinations may be equally effective. To form the denominator, a Subtraction primitive 172 subtracts the attrition (Att) input 174 from the number of weapons delivered (Nd) 176. The denominator value is passed to a Read Out primitive 178 to make this value available for display during simulation, an output connection 180 of the encasing block to make this value available for use through connection to other external blocks, one input to a Plotter primitive 182 that provides a time-based graph of population changes, and a Stop Message primitive 184 that ensures proper input as a denominator in the next primitive. To form the numerator, a Subtraction primitive 186 subtracts the units of resource available (NOT RFI) generated by a Resource Pool primitive 188 from the capacity of the resource pool provided by a Constant primitive 190. The Resource Pool primitive, a utility common ~~too~~ to many simulation software packages. It is used here for tracking hardware that is RFI. Hardware uses one unit of resource when RFI and relinquishes that unit whenever becoming not RFI, e.g., the hardware fails BIT or becomes due for scheduled maintenance. The advantage of using the Resource Pool primitive is that it is global and can be easily referenced and updated from anywhere within the model. The numerator value is passed to a Read Out primitive 192, a second input of Plotter primitive 182 and a Division primitive 194.

Please amend paragraph [0105] as follows:

Division primitive 194 divides the numerator by the denominator to calculate the stockpile availability A_s. An Equation primitive 196 contains an equation to ensure that the initial input to the Mean and Variance primitive 198 is one and never zero thereby producing input for a smooth curve to be plotted in the Plotter primitive 200. The Generator primitive 202 produces one dynamic object at a user specified time. At that time this object is briefly delayed in Delay primitive 204. This Delay primitive sends a positive signal from its "T" connector to reset the Mean and Variance primitive 198, effectively resetting the running average at that user specified time input to primitive 202. Running averages of A_s are typically reset (old observations discarded) after delivery ramp-up. The dynamic object is then discarded in the Exit primitive 206. The plotter and display field from this block can be used to provide plotted or scalar A_s statistics ~~to as~~ as part of an output interface of the model, or this common block.